

IN THE CLAIMS:

Claim 1 (Currently Amended): ~~On a computer having a memory, a~~ method for debugging a program on a computer ~~having a thread of execution, the method~~ comprising:

loading a debugger into a thread of execution of the program being debugged ~~independent of a breakpoint~~; and

running the debugger in the thread of execution being debugged to debug the program.

Claim 2 (Original): The method of claim 1, wherein the program comprises at least one object, the method further comprises calling an interface of the object via the debugger.

Claim 3 (Original): The method of claim 2, wherein the program is executing on a first computer and the object is located on a second computer that is in communication with the first computer, the method further comprising calling a proxy interface via the debugger, wherein the proxy interface is located on the first computer and has a pointer to the object.

Claim 4 (Original): The method of claim 2, further comprising: creating a socket for communicating with the debugger; and sending commands through the socket to the debugger for conversion into function calls to the object interface.

Claim 5 (Currently Amended): The method of claim 2, wherein the object is a COM object or a DCOM object.

Claim 6 (Canceled).

Claim 7 (Original): The method of claim 1, wherein the program executes within a process defined within the memory, the method further comprising: establishing communication with a console module located outside of the process; receiving a

command from the console; and converting the command into functions calls to the object interface.

Claim 8 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 1.

Claim 9 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 2.

Claim 10 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 3.

Claim 11 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 4.

Claim 12 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 5.

Claim 13 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 6.

Claim 14 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 7.

Claim 15 (Currently Amended): ~~On a computer having a memory, a~~ method for debugging a program on a computer, the program having a thread of execution within the memory, the thread being associated with context data for describing the context of the thread, the method comprising:

halting the a thread of execution of the program being debugged, wherein the thread is associated with context data describing context of the thread; independent of a breakpoint;

obtaining a pointer to an interface of an object of the program from the context data; and

referencing the pointer to make function calls manually to the interface from within the context of the thread being debugged.

Claim 16 (Original): The method of claim 15, wherein the program executes within a process defined within the memory, the method further comprising: establishing communication with a console module located outside of the process; receiving a command from the console; and converting the command into functions calls to the object interface.

Claim 17 (Original): The method of claim 16, wherein the establishing step comprises: creating a socket within the context of the thread; and communicating with the console module via the socket.

Claim 18 (Currently Amended): The method of claim 15, wherein the object is a COM object or a DCOM object.

b3
Claim 19 (Canceled).

Claim 20 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 15.

Claim 21 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 16.

Claim 22 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 17.

Claim 23 (Currently Amended): On a computer having a memory, a system for communicating with an object that is accessible by a program having a thread of execution within the memory, the thread being associated with context data for describing the context of the thread, the system comprising:

a debugger module operating within the context of the thread of the program being debugged;

a socket accessible by the debugger module for sending and receiving messages;
and

a console module operating outside of the context of the thread for receiving a command from a user and sending the command to the debugger via the socket, wherein the debugger converts the command into a function call to the object.

Claim 24 (Original): The system of claim 23, wherein the debugger module is one of a plurality of debugger modules, the system further comprising: a multiplexor module for multiplexing commands from the console module to each of the plurality of debugger modules.

Claim 25 (Original): The system of claim 24, wherein each of the plurality of debugger modules is associated with a socket session, and the commands entered at the console module are multiplexed to the appropriate debugger module based on the socket session.

63
Claim 26 (Currently Amended): A method for debugging a mission-critical program, the method comprising:

accessing ~~the~~ a computer on which the program is running via a public network;

halting a thread of execution of the program;

allowing other threads of execution of the program to continue;

loading a debugger into the ~~program's~~ halted thread of execution of the program;

and

running the debugger in the ~~program's~~ halted thread of execution to debug the program.

Claim 27 (Original): The method of claim 26, wherein the mission-critical program is executing on a web server.

Claim 28 (Previously Presented): The method of claim 26, wherein the mission-critical program is an electronic commerce program.

Claim 29 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 26.

Claim 30 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 27.

Claim 31 (Original): A computer-readable medium having stored thereon computer-executable instructions for performing the method of claim 28.

Claim 32 (Currently Amended): On a computer having a memory, a method for debugging a program running on a distributed environment ~~having a thread of execution~~, the method comprising:

halting ~~the a~~ thread of execution of the program being debugged, wherein the thread is associated with context data describing context of the thread;

creating a call-frame that will behave as if called from the program at the location at which the thread was halted;

loading a debugger into the halted thread of execution of the program; and

running the debugger in the thread of execution to debug the program, the debugger calling objects from the context of the thread of the program.

Claim 33 (Currently Amended): A method for debugging a mission-critical system that facilitates electronic commerce purchases for a large number of consumers over a public network that receives and fulfills the commerce purchases via an internet information server (IIS) module, the method comprising:

remotely accessing the system while the IIS module fulfills the commerce purchases;

executing a thread of an active server page (ASP) module being debugged while a plurality of other threads run concurrently with the thread of the ASP module; ~~and~~

executing a separate diagnostic thread to service the ASP module ~~independent of halting the ASP module;~~

loading a debugger into the thread of the ASP module being debugged; and

running the debugger in the thread being debugged to debug the ASP module.

B3 Claim 34 (Previously Presented): The method of claim 33 further comprising
accessing an object running on a browser operated by one of the consumers over the
public network.
